

**TITLE:** *Method for morphing geometric shapes based upon direction map*

This application was preliminarily rejected pursuant to Article 63 of the Korean Patent Law based on the following reason. Should there be any opinion against this action, please file a written argument by January 26, 2005. (You can apply for 1-month extension per each case, and we do not notify you of the confirmation for such term extension.)

**[REASON]**

This invention described in claims 1, 2 and 23 can be easily invented by those skilled in the art as pointed out below. Accordingly, the above-identified patent application cannot be registered pursuant to Article 29, Paragraph 2 of the Korean Patent Law.

**[BELOW]**

Claims 1, 2 and 23 of the present invention is directed to a method for morphing geometric shapes based upon a direction map and a recording medium thereof, while a reference 1 (Japanese Patent Laid-Open No. 2001-175881) discloses a mehod for correcting a satellite image which is distorted geometrically.

The geometric shape morphing method of the present invention includes the steps of extracting a direction map of each geometric shape; merging the direction maps; performing group-based scaling of the merged direction map; generating a polygon by using an inverse function of the merged direction map; and determining the number of in-betweens to be generated, and determining a group-based scaling method and a speed controlling method according to the number of in-betweens. It is understood that the structure for morphing a geometric shape can be easily invented by those skilled in the art from the structure of the reference 1 which includes a target shape outline extracting unit; an outline smoothing unit; a planarizing unit for removing self-intersection; a sequence generating unit; and a morphing executing unit.

*[Attachment] JP Laid-Open No. 2001-175881  
(Published on June 29, 2001)*

## PATENT ABSTRACTS OF JAPAN

(11)Publication number : 2001-175881

(43)Date of publication of application : 29.06.2001

(51)Int.Cl.

G06T 13/00

(21)Application number : 11-353720

(71)Applicant : INTERNATL BUSINESS MACH CORP &lt;IBM&gt;

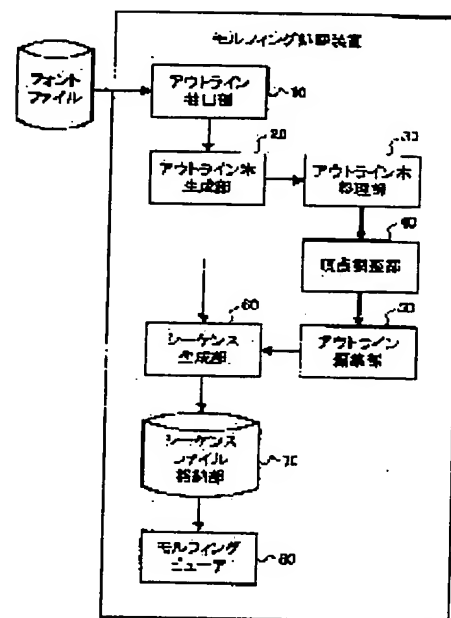
(22)Date of filing : 13.12.1999

(72)Inventor : AONO MASAKI

(54) MORPHING PROCESSOR, MORPHING METHOD, STORAGE MEDIUM, PROGRAM TRANSMISSION DEVICE, AND MOVING-PICTURE GENERATING DEVICE

(57)Abstract:

PROBLEM TO BE SOLVED: To enable an image process by morphing for outlines of various figures without causing mixture between vector spaces.  
 SOLUTION: The morphing processor which performs morphing is equipped with an outline extraction part 10 which inputs two object figures of a morphing process and extracts sets of outlines of the object figures, an outline tree generation part 20 and an outline tree rearrangement part 30 which erase outlines except one specific outline from the sets of outlines extracted by the object figures, an outline editing part 50 which performs a smoothing process for the two remaining outlines by the object figures without causing self-mixture at the time of the morphing, and a sequence generation part 60 and a morphing viewer 80 which perform morphing while reflecting the process of the smoothing process.



## LEGAL STATUS

[Date of request for examination] 27.12.1999

[Date of sending the examiner's decision of rejection]

[Kind of final disposal of application other than the examiner's decision of rejection or application converted registration]

[Date of final disposal for application]

[Patent number] 3316758

[Date of registration] 14.06.2002

[Number of appeal against examiner's decision of rejection]

[Date of requesting appeal against examiner's decision of rejection]

[Date of extinction of right]

Copyright (C); 1998,2003 Japan Patent Office

(19) 日本国特許庁 (J P)

(12) 公開特許公報 (A)

(11) 特許出願公開番号

特開2001-175881

(P2001-175881A)

(43) 公開日 平成13年6月29日 (2001. 6. 29)

(51) Int.Cl.

G 0 6 T 13/00

識別記号

F I

G 0 6 F 15/62

キーワード(参考)

3 4 0 A 5 B 0 5 0

審査請求 有 請求項の数22 O L (全 24 頁)

(21) 出願番号

特願平11-353720

(22) 出願日

平成11年12月13日 (1999. 12. 13)

(71) 出願人 390009531

インターナショナル・ビジネス・マシーンズ・コーポレーション

INTERNATIONAL BUSIN  
ESS MACHINES CORPO  
RATION

アメリカ合衆国10504、ニューヨーク州

アーモンク (番地なし)

(74) 代理人 100086243

弁理士 坂口 博 (外3名)

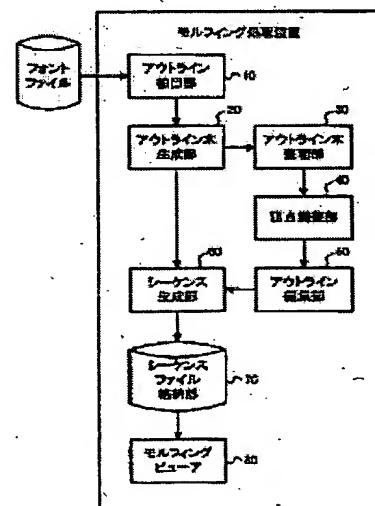
最終頁に続く

(54) 【発明の名称】 モルフィング処理装置、モルフィング方法、記憶媒体、プログラム伝送装置、および動画作成装置

(57) 【要約】

【課題】 多種多様な図形の輪郭線に対してベクトル空間で自己交錯を起こさずにモルフィングによる画像処理ができるようにする。

【解決手段】 モルフィングを行うモルフィング処理装置において、モルフィング処理の対象となる二つの対象図形を入力し、この対象図形のアウトラインの集合を抽出するアウトライン抽出部10と、この対象図形ごとに抽出されたアウトラインの集合のうち、それぞれ特定の一つのアウトラインのみを残して他のアウトラインを消滅させるアウトライン木生成部20及びアウトライン木整理部30と、この残された対象図形ごとの二つのアウトラインに対して、モルフィングの際に自己交錯を起こさないように平滑化処理を施すアウトライン編集部50と、この平滑化処理の経過を反映させてモルフィングを実行するシーケンス生成部60及びモルフィング・ビューア80とを備える。



【特許請求の範囲】

【請求項 1】 二つの図形を一方から他方へ滑らかに変形させるモルフィングを行うモルフィング処理装置において、

モルフィング処理の対象となる二つの対象図形を入力し、当該対象図形のアウトラインの集合を抽出するアウトライン抽出部と、

前記アウトライン抽出部により前記対象図形ごとに抽出された前記アウトラインの集合のうち、それぞれ特定の一つのアウトラインのみを残して他のアウトラインを消滅させるアウトライン整理部と、

前記アウトライン整理部により残された前記対象図形ごとの二つのアウトラインに対して、モルフィングの際に自己交錯を起こさないように平滑化処理を施す平滑化処理部と、

前記平滑化処理部による平滑化処理の経過を反映させてモルフィングを実行するモルフィング実行部とを備えることを特徴とするモルフィング処理装置。

【請求項 2】 前記アウトライン整理部は、前記アウトライン抽出部により前記対象図形ごとに抽出された前記アウトラインに関して、当該アウトラインに対応したノードを持ち、当該アウトライン相互の関係を反映させたアウトライン木を生成するアウトライン木生成部と、予め設定された規則に基づいて、前記アウトライン木生成部により前記対象図形ごとに生成された前記アウトライン木におけるノードのうち、特定の一つのノードのみを残して他のノードを削除するアウトライン木整理部とを備え、

前記平滑化処理部は、前記アウトライン整理部の処理において残された前記ノードに対応する前記アウトラインを対象として、平滑化処理を行うことを特徴とする請求項 1 に記載のモルフィング処理装置。

【請求項 3】 前記アウトライン木生成部は、前記アウトライン抽出部により抽出された前記対象図形のアウトラインを入力し、前記対象図形を構成する各アウトラインごとに当該アウトラインを囲むバウンディング・ボックスを設定し、当該バウンディング・ボックスの相互関係に基づいてアウトライン木を生成することを特徴とする請求項 2 に記載のモルフィング処理装置。

【請求項 4】 前記アウトライン抽出部は、前記対象図形のアウトラインに曲線が含まれている場合に、当該曲線部分を細分化して線分近似することを特徴とする請求項 1 に記載のモルフィング処理装置。

【請求項 5】 前記アウトライン抽出部は、前記対象図形としてアウトライン・フォントによる文字を入力し、当該文字のアウトラインを抽出することを特徴とする請求項 1 に記載のモルフィング処理装置。

【請求項 6】 前記アウトライン抽出部は、前記対象図形としてトゥルータイプフォントによる文字を入力し、当該文字のアウトラインを抽出することを特徴とする請求項 1 に記載のモルフィング処理装置。

【請求項 7】 前記平滑化処理部は、前記平滑化処理として、前記アウトラインにラプシアン・スムージングを施すことを特徴とする請求項 1 に記載のモルフィング処理装置。

【請求項 8】 前記モルフィング実行部は、前記平滑化処理部による前記平滑化処理の過程で生成されたアウトラインを、タイムパラメータに沿って並べたシーケンスファイルを生成するシーケンス生成部と、前記シーケンス生成部により生成された前記シーケンスファイルに基づいてモルフィングを実行するモルフィング実行部とを備えることを特徴とする請求項 1 に記載のモルフィング処理装置。

【請求項 9】 モルフィングにより二つの図形を一方から他方へ滑らかに変形させるモルフィング方法において、モルフィング処理の対象となる二つの対象図形のアウトラインを識別別として抽出するステップと、前記対象図形ごとに抽出されたアウトラインに関して、当該アウトラインに対応したノードを持ち、当該アウトライン相互の関係を反映させたアウトライン木を生成するステップと、

前記対象図形ごとに生成された前記アウトライン木におけるノードのうち、特定の一つのノードのみを残して他のノードを削除するステップと、

前記対象図形ごとの前記アウトライン木において残された前記ノードに対応する二つの前記アウトラインにおける頂点の数を一致させるステップと、

頂点の数を一致させた二つの前記アウトラインに対して、モルフィングの際に自己交錯を起こさないように平滑化処理を施すステップと、

前記平滑化処理の過程で生成されたアウトラインに関して、タイムパラメータに沿ったシーケンスファイルを生成するステップと、

生成された前記シーケンスファイルに基づいてモルフィングを実行し、ディスプレイ表示を行うステップとを含むことを特徴とするモルフィング方法。

【請求項 10】 前記アウトライン木を生成するステップは、

前記対象図形を構成する各アウトラインごとに当該アウトラインを囲むバウンディング・ボックスを設定するステップと、

前記対象図形の全体を囲むバウンディング・ボックスを生成し、当該バウンディング・ボックスに対応させてアウトライン木のルート・ノードを生成するステップと、

前記アウトラインごとに生成された前記バウンディング・ボックスを一つずつ選択し、選択された前記バウンディング・ボックスの前記アウトラインに対応する子ノードを生成し、前記ルート・ノードの下に挿入するステップとを含む。

前記子ノードを挿入するステップは、生成された所定の子ノードに対応する前記アウトラインを囲む前記バウンディング・ボックスが、既に前記ルート・ノードの下に挿入されている他の子ノードに対応する前記アウトラインを囲む前記バウンディング・ボックスに完全に含まれる場合に、当該子ノードを当該他の子ノードの下に再帰的に挿入するステップを含むことを特徴とする請求項 9 に記載のモルフィング方法。

【請求項 11】 前記アウトライン木のノードを削減するステップは、所定の規則に基づいてどの前記ノードを残すかを決定するステップと、

他の前記ノードに対応する前記アウトラインの各頂点を任意の一点に向けて移動させて、最終的にディスプレイに表示しない点とするステップとを含むことを特徴とする請求項 9 に記載のモルフィング方法。

【請求項 12】 前記アウトラインの頂点の数を一致させるステップは、前記二つのアウトラインにおける頂点列を、特定の位置を先頭にし、かつ一定の方向に前記アウトラインを周回するように並べるステップと、並べ替えられた前記頂点列における各頂点の座標値を、一定の範囲内の値を取るパラメータで表現するステップと、

前記二つのアウトラインにおけるパラメータ化された頂点列を比較し、一方のアウトラインにおけるパラメータ化された頂点を用いて他方のアウトラインに頂点を線形補間する操作を相互に行うステップとを含むことを特徴とする請求項 9 に記載のモルフィング方法。

【請求項 13】 前記パラメータ化された頂点列を用いて二つの前記アウトラインに頂点を線形補間するステップにおいて、

前記パラメータに基づいて、一方のアウトラインに含まれる頂点に対応する点を他方のアウトラインに追加する操作を相互に行うことを特徴とする請求項 12 に記載のモルフィング方法。

【請求項 14】 前記平滑化処理を行うステップにおいて、前記平滑化処理として、前記アウトラインに対してラブラシアン・スムージングを任意の回数繰り返して施し、モルフィングを行っても自己交錯が起こらなくなった時点で前記ラブラシアン・スムージングの適用を止めることを特徴とする請求項 9 に記載のモルフィング方法。

【請求項 15】 前記平滑化処理を行うステップは、前記頂点の数を一致させた二つの前記アウトラインに対して試行的にモルフィングを行って自己交錯が起こるかどうを確認するステップと、前記モルフィングの試行を行った結果、自己交錯が起こった場合に、前記二つのアウトラインに対し、平滑化処理として任意の回数のラブラシアン・スムージングを実

行するステップとを含み、

前記二つのステップを、前記モルフィングの試行を行った結果、自己交錯が起こらなくなるまで再帰的に繰り返すことを特徴とする請求項 9 に記載のモルフィング方法。

【請求項 16】 コンピュータに実行させるプログラムを当該コンピュータの入力手段が読取可能に記憶した記憶媒体において、

前記プログラムは、モルフィング処理の対象となる二つの対象図形を入力する処理と、入力された二つの前記対象図形のアウトラインを線分列として抽出する処理と、

前記対象図形ごとに抽出されたアウトラインに関して、当該アウトラインに対応したノードを持ち、当該アウトライン相互の関係を反映させたアウトライン木を生成する処理と、

前記対象図形ごとに生成された前記アウトライン木におけるノードのうち、特定の一つのノードのみを残して他のノードを削除する処理と、

前記対象図形ごとの前記アウトライン木において残された前記ノードに対応する二つの前記アウトラインにおける頂点の数を一致させる処理と、

頂点の数を一致させた二つの前記アウトラインに対して、モルフィングの際に自己交錯を起こさないように平滑化処理を施す処理と、

前記平滑化処理の過程で生成されたアウトラインに関して、タイムパラメータに沿ったシーケンスファイルを生成する処理と、

生成された前記シーケンスファイルに基づいてモルフィングを行い、ディスプレイ表示を行う処理とを前記コンピュータに実行させることを特徴とする記憶媒体。

【請求項 17】 コンピュータに、モルフィング処理の対象となる二つの対象図形を入力する処理と、入力された二つの前記対象図形のアウトラインを線分列として抽出する処理と、前記対象図形ごとに抽出されたアウトラインに関して、当該アウトラインに

対応したノードを持ち、当該アウトライン相互の関係を反映させたアウトライン木を生成する処理と、前記対象図形ごとに生成された前記アウトライン木におけるノードのうち、特定の一つのノードのみを残して他のノードを削除する処理と、前記対象図形ごとの前記アウトライン木において残された前記ノードに対応する二つの前記

アウトラインにおける頂点の数を一致させる処理と、頂点の数を一致させた二つの前記アウトラインに対して、モルフィングの際に自己交錯を起こさないように平滑化処理を施す処理と、前記平滑化処理の過程で生成された

アウトラインに関して、タイムパラメータに沿ったシーケンスファイルを生成する処理と、生成された前記シーケンスファイルに基づいてモルフィングを行い、ディス

プレイ表示を行う処理とを実行させるプログラムを記憶する記憶手段と、前記記憶手段から前記プログラムを読み出して当該プログラムを送信する送信手段とを備えたことを特徴とするプログラム伝送装置。

【請求項 18】 演算装置と、当該演算装置による演算結果を画像表示するディスプレイ装置とを備え、

前記演算装置は、

任意の  $m$  個の図形を要素とする図形列と任意の  $n$  個の図形を要素とする図形列とを処理対象として入力し、当該二つの図形列の要素を 1 対 1 で対応させると共に、前記二つの図形列における要素の数が同一でない場合は、要素が不足している方の前記図形列に、不足している要素の数だけ前記ディスプレイ装置に表示されない円を設けて、他の前記図形列における余っている要素に対応させ、

対応付けられた前記要素ごとにモルフィングを行うことにより、一方の図形列から他方の図形列へ滑らかに変形させることを特徴とするモルフィング処理装置。

【請求項 19】 二つの文字を一方から他方へ滑らかに変形させるモルフィングを行うモルフィング処理装置において、

モルフィング処理の対象となる二つの対象文字を入力し、当該対象文字のアウトラインの集合を抽出するアウトライン抽出部と、

前記アウトライン抽出部により前記対象文字ごとに抽出された前記アウトラインの集合のうち、それぞれ特定の一つのアウトラインのみを残して他のアウトラインを消滅させるアウトライン整理部と、

前記アウトライン整理部により残された前記対象文字ごとの二つのアウトラインに対して、モルフィングの際に自己交錯を起こさないように平滑化処理を施す平滑化処理部と、

前記平滑化処理部による平滑化処理の経過を反映させてモルフィングを実行するモルフィング実行部とを備えることを特徴とするモルフィング処理装置。

【請求項 20】 二つのフォントを一方から他方へ滑らかに変形させるモルフィングを行うモルフィング処理装置において、

モルフィング処理の対象となる二つの対象フォントを入力し、当該対象フォントのアウトラインの集合を抽出するアウトライン抽出部と、

前記アウトライン抽出部により前記対象フォントごとに抽出された前記アウトラインの集合のうち、それぞれ特定の一つのアウトラインのみを残して他のアウトラインを消滅させるアウトライン整理部と、

前記アウトライン整理部により残された前記対象フォントごとの二つのアウトラインに対して、モルフィングの際に自己交錯を起こさないように平滑化処理を施す平滑化処理部と、

前記平滑化処理部による平滑化処理の経過を反映させてモルフィングを実行するモルフィング実行部とを備えることを特徴とするモルフィング処理装置。

【請求項 21】 二つの図形を一方から他方へ滑らかに変形させるモルフィングを行うモルフィング処理装置において、

モルフィング処理の対象となる二つの対象図形に関して当該対象図形を形成するアウトラインの集合を入力し、当該アウトラインの各集合に対して、それぞれ特定の一つのアウトラインのみを残して他のアウトラインを消滅させるアウトライン整理部と、

前記アウトライン整理部により残された前記対象図形ごとの二つのアウトラインに対して、モルフィングの際に自己交錯を起こさないように平滑化処理を施す平滑化処理部と、

前記平滑化処理部による平滑化処理の経過を反映させてモルフィングを実行するモルフィング実行部とを備えることを特徴とするモルフィング処理装置。

【請求項 22】 二つの図形を一方から他方へ滑らかに変形させる動画を作成する動画作成装置において、画像処理の対象となる二つの対象図形を入力し、当該対象図形のアウトラインの集合を抽出するアウトライン抽出部と、

前記アウトライン抽出部により前記対象図形ごとに抽出された前記アウトラインの集合のうち、それぞれ特定の一つのアウトラインのみを残して他のアウトラインを消滅させるアウトライン整理部と、

前記アウトライン整理部により残された前記対象図形ごとの二つのアウトラインに対して、一方のアウトラインから他方のアウトラインへ変形する際に自己交錯を起こさないように平滑化処理を施す平滑化処理部と、

前記平滑化処理部による平滑化処理の経過を反映させて、前記二つの対象図形を一方から他方へ滑らかに変形させる動画を作成する画像処理部とを備えることを特徴とする動画作成装置。

#### 【発明の詳細な説明】

【0001】

【発明の属する技術分野】 本発明は、直線や曲線からなる文字または図形の輪郭に対してオブジェクト空間でモルフィング (morphing) 処理を行うモルフィング方法に関する。

【0002】

【従来の技術】 近年、コンピュータ・グラフィックスによる動画技術として、モルフィングを用いた画像処理が広く利用されている。モルフィングは、画像上に設定した制御点を動かすことにより画像が歪む効果を得て、2種類の図形を一方から他方へ滑らかに変形させる技術である。

【0003】 このモルフィング技術を用いて、ディスプレイ装置上に表示された文字を変形させることができ

は、例えばインターネットで情報配信に用いられるホームページのような視覚効果が重視される文書において、利用価値が高い。

【0004】ディスプレイ装置上に表示された文字をモルフィングにより変形させる従来技術が、文献1「Advanced Animation and Rendering Techniques」(Alan Watt, Mark Watt 著, Addison-Wesley, 1992, pp. 413)に開示されている。文献1には、ベジェ曲線(Bezier curve)で定義された文字に関し、ベジェ曲線の制御点を線形補間して変形する手法が記載されている。また、実際の動作例として、文字「E」から文字「Z」に変換する例が記載されている。

【0005】しかし、文献1に開示された技術は、ベジェ曲線だけで定義された文字を対象としており、True Type フォント等の一般的なアウトライン・フォントのように線分とベジェ曲線との組み合わせで構成された文字にそのまま応用することができず、汎用性を欠く。また、例示された文字「E」や文字「Z」のように輪郭線が単一(single outline)の文字しか扱えない。すなわち、文字「A」や文字「B」のように穴がある文字や漢字のように画数の多い文字など、複数の輪郭線(multiple outlines)で形成された文字を対象として使用することができない。

【0006】また、文字の変形のみを対象としたものではないが、2次元のアウトライン曲線をモルフィングにより変形させる従来技術が、例えば、文献2「A Physically Based Approach to 2-D Shape Blending」(T. W. Sederberg 他著, SIGGRAPH Proceeding, 1992, pp. 25-34)、文献3「2-D Shape Blending: An Intrinsic Solution to the Vertex Path Problem」(T. W. Sederberg 他著, SIGGRAPH Proceeding, 1993, pp. 15-18)、文献4「A Fuzzy Approach to Digital Image Warping」(Y. Zhang 著, 1996, July, pp. 34-41)、文献5「Self-intersection elimination in metamorphosis of two-dimensional curves」(T. Samoilov, G. Elber 著, The Visual Computer, 1988, Vol. 14, pp. 415-428)等の論文に開示されている。

【0007】文献2及び文献3に開示された技術は、リニア曲線またはポリラインなどの直線(辺)を組み合わせて形成された輪郭線において、隣接する辺のなす角度及び長さをパラメータとして変形を行うアルゴリズムを用いている。しかし、当該従来技術は、文字のように凹凸の多い図形のモルフィングは困難である。文献4に開示された技術は、多角形の2次元画像を歪ませてイメージ空間でモルフィングさせる方法である。しかし、当該従来技術は、イメージ空間で処理を行うため、輪郭がギザギザになる(jaggy contour)場合がある。また、図形の拡大縮小ができない(任意のスケールでモルフィングできない)。文献5は、モルフィングの途中で自己交錯(self-intersection)を起こさないようにする二つ

のアルゴリズムを提案する。しかし、これらのアルゴリズムは、二つの2次元閉曲線が位相的に類似していることが前提となっているため、変形の際に自己交錯を回避できる図形が制限される。

【0008】

【発明が解決しようとする課題】上述したように、従来のモルフィング技術は、文字のような複雑な図形の輪郭を変形するために用いることが困難であった。すなわち、文献4の従来技術のようにイメージ空間で処理を行う場合、輪郭がギザギザになる場合があり、見栄えが良くない。また、任意のスケールでモルフィングできないため、視覚効果が制限されてしまう。

【0009】イメージ空間ではなくオブジェクト空間でベクトルを用いて処理を行う場合であっても、文献1の従来技術のように単一の輪郭線で表現できる図形しか扱えないものや、文献2、文献3の従来技術のように凹凸の多い図形には使用できないものなど、使用対象が制限され、多種多様な図形に対して汎用的に用いることができない。

【0010】さらに、図形の輪郭をモルフィングにより変形する場合、自己交錯をいかにして回避するかということが問題となる。これは、輪郭内部を特定の色で塗りつぶすような場合、輪郭線が自己交錯を起こしていると、うまく処理できない場合があるためである。しかし、かかる問題の解決を試みる文献5の従来技術であっても、自己交錯を回避できる図形が制限されるため、汎用性に欠ける。

【0011】本発明は以上のような技術的課題を解決するためになされたものであって、多種多様な図形の輪郭線に対してベクトル空間でモルフィングによる画像処理ができるようにすることを目的とする。具体的には、複数の輪郭線で形成された図形に対してもモルフィングを行えるようにする。また、凹凸の多い図形に対しても自己交錯を起こさなくモルフィングを行えるようにする。

【0012】更に、コンピュータのディスプレイ装置上に表示された文字や文字列に対してもモルフィングによる画像処理ができるようにすることを他の目的とする

【0013】

【課題を解決するための手段】かかる目的のもと、本発明は、二つの図形を一方から他方へ滑らかに変形させるモルフィングを行うモルフィング処理装置において、モルフィング処理の対象となる二つの対象図形を入力し、この対象図形のアウトラインの集合を抽出するアウトライン抽出部と、このアウトライン抽出部により対象図形ごとに抽出されたアウトラインの集合のうち、それぞれ特定の一つのアウトラインのみを残して他のアウトラインを消滅させるアウトライン整理部と、このアウトライン整理部により残された対象図形ごとの二つのアウトラインに対して、モルフィングの際に自己交錯を起こさな

いように平滑化処理を施す平滑化処理部と、この平滑化処理部による平滑化処理の経過を反映させてモルフィングを実行するモルフィング実行部とを備えることを特徴としている。

【0014】ここで、アウトライン整理部は、アウトライン抽出部により対象図形ごとに抽出されたアウトラインに関して、このアウトラインに対応したノードを持ち、このアウトライン相互の関係を反映させたアウトライン木を生成するアウトライン木生成部と、予め設定された規則に基づいて、アウトライン木生成部により対象図形ごとに生成されたアウトライン木におけるノードのうち、特定の一つのノードのみを残して他のノードを削除するアウトライン木整理部とを備え、平滑化処理部は、アウトライン整理部の処理において残されたノードに対応するアウトラインを対象として、平滑化処理を行うことを特徴としている。このようにすれば、アウトライン木のノードを指定することにより、アウトライン木の構成を考慮して所望のアウトラインを残すことができる点で好ましい。また、後述するように、アウトライン木のノードにはアウトラインを囲むバウンディング・ボックスについての情報が格納されるが、バウンディング・ボックスの面積が最大であるノードを残すといった処理も可能である。さらにここで、アウトライン木生成部は、アウトライン抽出部により抽出された対象図形のアウトラインを入力し、この対象図形を構成する各アウトラインごとに当該アウトラインを囲むバウンディング・ボックスを設定し、このバウンディング・ボックスの相互関係に基づいてアウトライン木を生成することを特徴としている。このようにすれば、対象図形を構成するアウトライン相互の関係を明瞭にすることができる点で好ましい。また、バウンディング・ボックスの形状は任意に選択することができるが、文字を対象図形とする場合には、長方形とするのが扱いやすい。

【0015】さらにアウトライン抽出部は、対象図形のアウトラインに曲線が含まれている場合には、この曲線部分を細分化して線分近似することを中心としている。このようにすれば、アウトラインを多角形として扱うことができ、処理を一般化させることができるため、対象図形の形状によらない処理を行うことが可能となる点で優れている。さらにアウトライン抽出部は、対象図形としてアウトライン・フォントによる文字を入力し、この文字のアウトラインを抽出することを特徴としている。これにより、テキストを表示した文書に対して大きな視覚的効果を与えることができる。尚、ドット・フォントに対しても、前処理としてそのアウトラインを抽出することができれば、モルフィングの対象とすることができる。さらにアウトライン抽出部は、対象図形としてユーザタイプフォントによる文字を入力し、この文字のアウトラインを抽出することを特徴としている。これにより、パーソナルコンピュータにおいて広く利用されてい

るフォントに対してモルフィングを行うことができるため、汎用性が増す。

【0016】さらに平滑化処理部は、平滑化処理として、アウトラインに対してラブラシアン・スムージングを施すことを特徴としている。ラブラシアン・スムージングは平滑化処理の一例であり、他の手法を用いることも可能である。

【0017】さらにモルフィング実行部は、平滑化処理部による平滑化処理の過程で生成されたアウトラインを、タイムパラメータに沿って並べたシーケンスファイルを生成するシーケンス生成部と、このシーケンス生成部により生成されたシーケンスファイルに基づいてモルフィングを実行するモルフィング実行部とを備えることを特徴としている。このようにすれば、シーケンスファイルを送ることにより自己交錯が発生しないモルフィングを実現できるのみならず、シーケンスファイルを逆に送ることにより、反対方向のモルフィングも容易に実現できる点で優れている。なお、平滑化処理を行わなくてもモルフィングの際に自己交錯が発生しない場合は、シーケンスファイルとしては、アウトライン整理部にて残された二つのアウトラインだけで構成することができる。

【0018】また、本発明は、モルフィングにより二つの図形を一方から他方へ滑らかに変形させるモルフィング方法において、モルフィング処理の対象となる二つの対象図形のアウトラインを線分列として抽出するステップと、この対象図形ごとに抽出されたアウトラインに関して、このアウトラインに対応したノードを持ち、このアウトライン相互の関係を反映させたアウトライン木を生成するステップと、この対象図形ごとに生成されたアウトライン木におけるノードのうち、特定の一つのノードのみを残して他のノードを削除するステップと、対象図形ごとのアウトライン木において残されたノードに対応する二つのアウトラインにおける頂点の数を一致させるステップと、この頂点の数を一致させた二つのアウトラインに対して、モルフィングの際に自己交錯を起こさないように平滑化処理を施すステップと、この平滑化処理の過程で生成されたアウトラインに関して、タイムパラメータに沿ったシーケンスファイル生成するステップと、この生成されたシーケンスファイルに基づいてモルフィングを実行し、ディスプレイ表示を行うステップとを含むことを特徴としている。

【0019】ここで、アウトライン木を生成するステップは、対象図形を構成する各アウトラインごとに当該アウトラインを囲むバウンディング・ボックスを設定するステップと、対象図形の全体を囲むバウンディング・ボックスを生成し、このバウンディング・ボックスに対応させてアウトライン木のルート・ノードを生成するステップと、アウトラインごとに生成されたバウンディング・ボックスを一つずつ選択し、この選択されたバウンデ



ィング・ボックスのアウトラインに対応する子ノードを生成し、ルート・ノードの下に挿入するステップとを含み、子ノードを挿入するステップは、生成された所定の子ノードに対応するアウトラインを囲むバウンディング・ボックスが、既にルート・ノードの下に挿入されている他の子ノードに対応するアウトラインを囲むバウンディング・ボックスに完全に含まれる場合に、この子ノードをこの他の子ノードの下に再帰的に挿入するステップを含むことを特徴としている。これにより、対象図形のアウトラインの幾何的な構造をアウトライン木の構成に反映させることができる点で好ましい。

【0020】さらにアウトライン木のノードを削減するステップは、所定の規則に基づいてどのノードを残すかを決定するステップと、他のノードに対応するアウトラインの各頂点を任意の一点に向けて移動させて、最終的にディスプレイに表示しない点とするステップとを含むことを特徴としている。これにより、不要なアウトラインが次第に消えていく様子を表示させることができる。アウトラインを収束させる点の位置は任意でよいが、バウンディング・ボックスの中心へ移動させると視覚的にバランスがよく、好ましい。

【0021】さらにアウトラインの頂点の数を一致させるステップは、二つのアウトラインにおける頂点列を、特定の位置を先頭にし、かつ一定の方向にこのアウトラインを巡回するように並べるステップと、並べ替えられたこの頂点列における各頂点の座標値を、一定の範囲内の値を取るパラメータで表現するステップと、二つのアウトラインにおけるパラメータ化された頂点列を比較し、一方のアウトラインにおけるパラメータ化された頂点を用いて他方のアウトラインに頂点を線形補間する操作を相互に行うステップとを含むことを特徴としている。さらにここで、パラメータ化された頂点列を用いて二つのアウトラインに頂点を線形補間するステップにおいて、パラメータに基づいて、一方のアウトラインに含まれる頂点に対応する点を他方のアウトラインに追加する操作を相互に行うことを特徴としている。このようにすれば、線形補間により得られた新たな頂点を二つのアウトラインのファイル間でマージソートする必要がない点で優れている。

【0022】さらに平滑化処理を行うステップにおいて、平滑化処理として、アウトラインに対してラプシアン・スムージングを任意の回数繰り返して施し、モルフィングを行っても自己交錯が起らなかった時点でラプシアン・スムージングの適用を止めることを特徴としている。ラプシアン・スムージングを途中で止めることにより、対象図形の形状に関わらず平滑化処理によってほぼ円形になってしまうことを回避できるため、視覚的に好ましい。

【0023】さらに平滑化処理を行うステップは、頂点の数を一致させた二つのアウトラインに対して試行的に

モルフィングを行って自己交錯が起るかどうかなを確認するステップと、モルフィングの試行を行った結果、自己交錯が起った場合に、この二つのアウトラインに対し、平滑化処理として任意の回数のラプシアン・スムージングを実行するステップとを含み、この二つのステップを、モルフィングの試行を行った結果、自己交錯が起らなくなるまで再帰的に繰り返すことを特徴としている。このようにすれば、自己交錯が起らなくなった時点で機械的にラプシアン・スムージングを止めることができる点で優れている。

【0024】また、本発明は、コンピュータに実行させるプログラムを当該コンピュータの入力手段が読取可能に記憶した記憶媒体において、このプログラムは、モルフィング処理の対象となる二つの対象図形を入力する処理と、入力された二つの対象図形のアウトラインを線分列として抽出する処理と、対象図形ごとに抽出されたアウトラインに関して、このアウトラインに対応したノードを持ち、このアウトライン相互の関係を反映させたアウトライン木を生成する処理と、この対象図形ごとに生成されたアウトライン木におけるノードのうち、特定の一つのノードのみを残して他のノードを削除する処理と、対象図形ごとのアウトライン木において残されたノードに対応する二つのアウトラインにおける頂点の数を一致させる処理と、この頂点の数を一致させた二つのアウトラインに対して、モルフィングの際に自己交錯を起こさないように平滑化処理を施す処理と、この平滑化処理の過程で生成されたアウトラインに関して、タイムパラメータに沿ったシーケンスファイルを生成する処理と、生成されたこのシーケンスファイルに基づいてモルフィングを行い、ディスプレイ表示を行う処理とをこのコンピュータに実行させることを特徴としている。このようにすれば、このプログラムをロードしたコンピュータにおいて、自己交錯の起らないモルフィングを実現できる点で好ましい。

【0025】また、本発明は、コンピュータに、モルフィング処理の対象となる二つの対象図形を入力する処理と、入力された二つの対象図形のアウトラインを線分列として抽出する処理と、この対象図形ごとに抽出されたアウトラインに関して、このアウトラインに対応したノードを持ち、このアウトライン相互の関係を反映させたアウトライン木を生成する処理と、この対象図形ごとに生成されたアウトライン木におけるノードのうち、特定の一つのノードのみを残して他のノードを削除する処理と、対象図形ごとのアウトライン木において残されたノードに対応する二つのアウトラインにおける頂点の数を一致させる処理と、この頂点の数を一致させた二つのアウトラインに対して、モルフィングの際に自己交錯を起こさないように平滑化処理を施す処理と、この平滑化処理の過程で生成されたアウトラインに関して、タイムパラメータに沿ったシーケンスファイルを生成する処理

と、生成されたこのシーケンスファイルに基づいてモルフィングを行い、ディスプレイ表示を行う処理とを実行させるプログラムを記憶する記憶手段と、この記憶手段からこのプログラムを読み出してこのプログラムを送信する送信手段とを備えたことを特徴としている。このようなプログラム伝送装置により、プログラムの提供形態としてCD-ROM等の記憶媒体を介することなく、顧客に対して本発明の技術を提供することが可能となる。

【0026】また、本発明は、演算装置と、当該演算装置による演算結果を画像表示するディスプレイ装置とを備え、この演算装置は、任意の $m$ 個の図形を要素とする図形列と任意の $n$ 個の図形を要素とする図形列とを処理対象として入力し、この二つの図形列の要素を1対1で対応させると共に、この二つの図形列における要素の数が同一でない場合は、要素が不足している方の図形列に、不足している要素の数だけディスプレイ装置に表示されない円を設けて、他の図形列における余っている要素に対応させ、この対応付けられた要素ごとにモルフィングを行うことにより、一方の図形列から他方の図形列へ滑らかに変形させることを特徴としている。これにより、多対多の図形列に対してモルフィングができるため、汎用性が高まる点で好ましい。

【0027】更に本発明によれば、次のようなモルフィング処理装置を提供することができる。すなわち、モルフィング処理の対象となる二つの対象文字を入力し、この対象文字のアウトラインの集合を抽出するアウトライン抽出部と、このアウトライン抽出部によりこの対象文字ごとに抽出されたアウトラインの集合のうち、それぞれ特定の一つのアウトラインのみを残して他のアウトラインを消滅させるアウトライン整理部と、このアウトライン整理部により残されたこの対象文字ごとの二つのアウトラインに対して、モルフィングの際に自己交錯を起こさないように平滑化処理を施す平滑化処理部と、平滑化処理部による平滑化処理の経過を反映させてモルフィングを実行するモルフィング実行部とを備えることを特徴としている。ここで、処理対象となる文字は、コンピュータや専用ワードプロセッサ機などで用いられる文字コードで特定されるような文字とすることができる。これにより、キーボードから打ち込んだ文字をそのまま用いてモルフィングさせることが可能となる。

【0028】さらにまた、本発明によれば、次のようなモルフィング処理装置を提供することができる。すなわち、モルフィング処理の対象となる二つの対象フォントを入力し、この対象フォントのアウトラインの集合を抽出するアウトライン抽出部と、このアウトライン抽出部によりこの対象フォントごとに抽出されたアウトラインの集合のうち、それぞれ特定の一つのアウトラインのみを残して他のアウトラインを消滅させるアウトライン整理部と、このアウトライン整理部により残されたこの対象フォントごとの二つのアウトラインに対して、モルフィ

ングの際に自己交錯を起こさないように平滑化処理を施す平滑化処理部と、この平滑化処理部による平滑化処理の経過を反映させてモルフィングを実行するモルフィング実行部とを備えることを特徴としている。このような構成とすることにより、異なるフォントで表示された文字や文字列に対してもモルフィングによる変形処理を行うことが可能となる。

【0029】また、本発明は、二つの図形を一方から他方へ滑らかに変形させるモルフィングを行うモルフィング処理装置において、モルフィング処理の対象となる二つの対象図形に関してこの対象図形を形成するアウトラインの集合を入力し、このアウトラインの各集合に対して、それぞれ特定の一つのアウトラインのみを残して他のアウトラインを消滅させるアウトライン整理部と、このアウトライン整理部により残されたこの対象図形ごとの二つのアウトラインに対して、モルフィングの際に自己交錯を起こさないように平滑化処理を施す平滑化処理部と、この平滑化処理部による平滑化処理の経過を反映させてモルフィングを実行するモルフィング実行部とを備えることを特徴としている。このような構成とすれば、アウトラインフォントのように自らアウトラインの情報を持つ対象図形から直接アウトラインの情報を取得して、モルフィングを行うことができる。

【0030】更にまた、本発明は、二つの図形を一方から他方へ滑らかに変形させる動画を作成する動画作成装置において、画像処理の対象となる二つの対象図形を入力し、この対象図形のアウトラインの集合を抽出するアウトライン抽出部と、このアウトライン抽出部によりこの対象図形ごとに抽出されたアウトラインの集合のうち、それぞれ特定の一つのアウトラインのみを残して他のアウトラインを消滅させるアウトライン整理部と、このアウトライン整理部により残されたこの対象図形ごとの二つのアウトラインに対して、一方のアウトラインから他方のアウトラインへ変形する際に自己交錯を起こさないように平滑化処理を施す平滑化処理部と、この平滑化処理部による平滑化処理の経過を反映させて、この二つの対象図形を一方から他方へ滑らかに変形させる動画を作成する画像処理部とを備えることを特徴としている。

【0031】

【発明の実施の形態】以下、添付図面に示す実施の形態に基づいてこの発明を詳細に説明する。図1は、本実施の形態におけるモルフィング処理装置の全体構成を説明するための図である。同図において、符号10はアウトライン抽出部であり、モルフィングによる画像処理の対象である図形（以下、対象図形と称す）からアウトライン（輪郭線）を抽出する。符号20はアウトライン木生成部であり、対象図形を構成する各アウトラインをノードとするアウトライン木を生成する。符号30はアウトライン木整理部であり、アウトライン木生成部20によ

り生成されたアウトライン木を整理して、一つのアウトライン木に対してルート・ノードと一つの子ノードのみを残す。符号40は頂点調整部であり、二つの対象図形に関して、アウトライン木整理部30の処理により残された一つの子ノードのアウトラインにおける頂点の数を一致させる。符号50はアウトライン編集部であり、頂点調整部40により頂点の数を調整されたアウトラインを、自己交錯が起らないように変形する。符号60はシーケンス生成部であり、アウトライン木生成部20からアウトライン編集部50までの各機能部による処理を反映させたアウトラインのシーケンスファイルを作成する。符号70は、シーケンスファイル格納部であり、シーケンス生成部60により生成されたシーケンスファイルを格納する。符号80は、モルフィング・ビューアであり、シーケンスファイル格納部70からアウトラインのシーケンスファイルを読み出し、当該シーケンスファイルに基づいてモルフィングを行い、ディスプレイ装置に動画を表示する。

【0032】アウトライン抽出部10は、対象図形を入力し、そのアウトラインを線分の組み合わせとして抽出する。対象図形の輪郭線に曲線が含まれている場合は、当該曲線部分を細分化(tessellation)して線分近似することにより、アウトラインを形成する線分列を得る。これにより、対象図形の輪郭線が多角形で近似されることとなる。また、各線分はベクトルで表現される。本実施の形態における対象図形の輪郭は単一のアウトラインで形成される必要はない。すなわち、図形中に穴がある場合や、いくつかの図形の組み合わせで構成される図形についても、一つの対象図形として扱うことができる。例えば、文字を対象図形とする場合、文字「B」は、外側を囲むアウトラインと、中の二つの穴を形成する二つのアウトラインとで形成された図形である。また、文字「杉」は、四つのアウトラインで形成された図形である。なお、対象図形は、モルフィングによる変形前と変形後の二つある。したがって、これら二つの対象図形に関してそれぞれアウトラインの抽出を行う。

【0033】アウトライン木生成部20は、アウトライン抽出部10により抽出された対象図形のアウトラインを入力し、当該対象図形を構成する各アウトラインごとに当該アウトラインを囲むバウンディング・ボックス(bounding box)を設定し、当該バウンディング・ボックスの相互関係に基づいてアウトライン木を生成する。このアウトライン木の生成は、対象図形が必ず一つ以上のアウトラインから構成されるという前提に基づいて行われる。なお、バウンディング・ボックスの形状は長方形が一般的であるが、他の形状であっても良い。以下の説明では、長方形のバウンディング・ボックスを設定するものとして説明する。アウトライン木は、ルート・ノードの下に各アウトラインに対応する子ノードを挿入することにより形成される。各子ノードは、対応するアウ

トラインに関して、後述するアウトライン木の整理や頂点の調整、アウトラインの編集等の処理に用いられる情報を格納する。例えば、図8に示すように、delete flag(アウトライン木の整理の際に削除するかどうかを設定するフラグ(整数))、num Point(アウトラインの頂点の数(整数))、num Child(自ノードの下に位置する子ノードの数(整数))、outline(アウトラインの各頂点における初期の座標値(実数配列))、deform outline(アウトラインの各頂点におけるモルフィング時に使用される座標値(実数配列))、xmin, xmax, ymin, ymax(バウンディング・ボックスの座標(実数))、child(自ノードの子ノードへのポインタ(ノードへのポインタ配列))、cx, cy(バウンディング・ボックスの中心の座標値(実数))等が格納される。図9に文字「B」を対象図形とした場合のアウトライン木を、図10に文字「杉」を対象図形とした場合のアウトライン木を示す。図9に示すように、文字「B」の穴のような上位のノードのバウンディング・ボックス中に完全に含まれるアウトラインを持つノードは、当該上位のノードの子供になる。これに対し、図10に示すように、文字「杉」複数のバウンディング・ボックスが並べて配置される関係にある場合は、各ノードが同じレベルに位置することとなる。

【0034】アウトライン木整理部30は、アウトライン木生成部20により生成されたアウトライン木のうち、一つの子ノードのみを残して他の子ノードを消滅させる。ノードの消滅は、消滅させる各ノードにおけるアウトラインの全ての頂点を、対応するバウンディング・ボックスの中心に向けて移動させることにより、実現する。図11は、全体のタイムパラメータを $t$ とし、時刻 $T_1$ から時刻 $T_2$ までの間にノード $k$ のアウトラインを保持する座標変数outline[k][2]がバウンディング・ボックスの中心座標(cx, cy)に移動させる方法をC言語風に記述したものである。この記述に基づき、ディスプレイ装置において(x, y)を時刻 $[T_1, T_2]$ の間に描画することで、ノード $k$ で示される各ノードのアウトラインが、対応するバウンディング・ボックスの中心に向かって収縮し、最終的にディスプレイ装置には表示されない点となって消滅する。以上の操作によって、アウトライン木が整理され、一つの対象図形に対して一つのアウトラインのみが残ることとなる。これによって、変形前後の二つの対象図形は、それぞれ一つの多角形で表現されることとなる。すなわち、アウトラインを構成する頂点が $m$ 個あるならば、 $m$ 角形となる。ここで、対象図形を形成する複数のアウトラインのうち、どのアウトラインを残すかについては、任意に定めることができる。一般には、アウトラインごとに設定されているバウンディング・ボックスの面積が最大のものを残すようにすれば、見栄えが良い。また、対象図形に応じて特徴的な箇所のアウトラインを残すようにしても良い。

例えば、漢字などの文字を対象図形とする場合は、対象図形となる文字の書き順を考慮して残すアウトラインを決定することもできる。

【0035】頂点調整部40は、アウトライン木整理部30の処理により残された一つのアウトラインを入力し、当該アウトラインにおける頂点の数を一致させる。この処理を行うのは、二つの対象図形からそれぞれ取り出された二つのアウトラインにおける頂点の数は異なるのが普通であるため、そのままではモルフィングを行うことができない場合が多いからである。二つのアウトラインにおける頂点の数を一致させる処理は、次のようにして行う。まず、アウトラインを構成する頂点列を、特定の位置の頂点を先頭にして、当該アウトライン上を一定の方向に巡回するように並べる。具体的には、例えば、Y座標の値が最大の頂点（複数ある場合はその中でX座標の値が最大の頂点）を先頭にして頂点列を並べる。これにより、以下の処理の開始位置をバウンディング・ボックスのほぼ右上に揃えておくことができ、かつ後述する自己交錯を起こしにくくすることができる。なお、開始位置をバウンディング・ボックスのほぼ右上に揃えたとしたの例示に過ぎず、任意の位置から開始するように設定することが可能である。

【0036】次に、各アウトラインの全周の長さを求め、各頂点の座標値を[0.0-1.0]のパラメータで表現する。ここで、開始点のパラメータが0.0であり、アウトラインを一周して戻ってきた同じ点のパラメータが1.0である。具体的には、アウトラインの全周の長さをLとし、頂点の数をmとしたとき、パラメータpは下記の数1で表される。

$$\text{【数1】} \quad p = \frac{\sum_{i=1}^m |D_i - D_{i-1}|}{L}$$

上式において、 $D_i$ はi番目の頂点の座標値( $x_i, y_i$ )であり、 $|D_i - D_{i-1}|$ はi-1番目の頂点とi番目の頂点との間の距離を表す。また、求めたパラメータpは、浮動小数点で計算され、一般に0.0と1.0以外二つのアウトライン上で一致するパラメータは発生しない。仮に一致するパラメータがあったとしても、後の計算には影響しない。

【0037】次に、一方のアウトラインにおけるパラメータ化された頂点を用いて他方のアウトラインに頂点を線形補間する操作を相互に行うことにより、二つのアウトラインにおける頂点の数を一致させる。線形補間の結果を各アウトラインの頂点列に反映させるには、線形補間により求められた座標値を各アウトラインの頂点列とは別のファイルに書き出した後、パラメータ空間において、パラメータをキーとしてマージソートを行う。また、線形補間により求められた座標値を別ファイルに書

き出す代わりに、次のような手法を取ることもできる。すなわち、一方のアウトライン(OL1と称す)のパラメータpiは、両端の0.0と1.0を除いて、一般に他方のアウトライン(OL2と称す)には現れない。そこで、仮に $p_{j-1} < p_i < p_j$ （ただし、 $p_{j-1}$ 及び $p_j$ はアウトラインOL2の頂点のパラメータ）としたとき、アウトラインOL2上の $D_{j-1}$ と $D_j$ との間に、新しい頂点として、下記の数2で表される頂点 $D_{new}$ を加える。なお、当該処理によって新しく追加される頂点 $D_{new}$ は、アウトラインOL2上に元々存在した $D_{j-1}$ と $D_j$ との間の線分上に設けられるため、正しくは頂点ではないが、アウトラインOL1の頂点と対応させるために頂点として扱う。言い換えれば、線分 $D_{j-1} - D_{new}$ と線分 $D_{new} - D_j$ とが平行であるものとして扱う。

$$\text{【数2】} \quad D_{new} = \frac{p_j - p_i}{p_j - p_{j-1}} D_{j-1} + \frac{p_i - p_{j-1}}{p_j - p_{j-1}} D_j$$

かかる処理を、アウトラインOL1上の各頂点について実行することによりアウトラインOL2上に新しい頂点を加える。さらに、同様にして、アウトラインOL2上の頂点のパラメータに基づいてアウトラインOL1上に新しい頂点を加えることにより、アウトラインOL1とアウトラインOL2の頂点の数が一致する。このような処理を行えば、ファイル間のマージソートを行う必要がない。

【0038】アウトライン編集部50は、頂点調整部40によって、頂点の数を一致させた二つのアウトラインに対し、モルフィングを行った際に自己交錯が起きないように、必要に応じて平滑化処理を行う。すなわち、頂点調整部40により頂点の数を一致させた段階のアウトラインを対象として単純にモルフィングを行うと、アウトラインの形状によっては自己交錯を起こしてしまう。特に凹凸の多い複雑な形状のアウトラインでは自己交錯を起こしやすい。変形の際に自己交錯が発生すると、単に見た目が悪いだけでなく、アウトラインの内部を特定の色で塗りつぶすような場合にうまく処理できない場合がある。そこで、二つのアウトライン間のモルフィングを行った際に自己交錯が起こる場合には、自己交錯が起こらないような形に変形する。このために行う処理が平滑化処理である。自己交錯を起こさなくモルフィングを行うことができる条件は、アウトラインの形状が全体的に外側へ緩やかな凸状になっていることである。例えば円や四角形や三角形といった単純な形状が相当する。これに対し、任意の対象図形では、アウトライン上に急激な凹凸部分があるような複雑な形状の場合がある。特に漢字などの文字を対象図形とする場合には複雑

な凹凸を持つ形状である場合が多い。そこで、当該アウトラインに対して、かかる凹凸を取り除くための平滑化処理を行う。平滑化処理の具体例としては、信号処理で用いられるラプラシアン・スムージング (Laplacian smoothing) を用いることができる。この手法によれば、各頂点に対して平滑化処理による座標値  $D_i'$  を得るため、下記の数3の演算を行う。

【数3】

$$D_i' = D_i + \alpha \times \Delta D_i$$

$$\Delta D_i = \frac{1}{2}(D_{i-1} - D_i) + \frac{1}{2}(D_{i+1} - D_i)$$

上式において、変数  $\alpha$  は平滑化の速度を制御するパラメータであり、通常は0.0から1.0までの値を取る。このラプラシアン・スムージングを複数回適用することにより、アウトラインが次第に平滑化され、凹部が取り除かれることとなる。

【0039】ここで、ラプラシアン・スムージングを繰り返し適用していくと、アウトラインは最終的に円に近づいていく。したがって、変形前後の二つの対象図形がどのような形状であっても、各対象図形のアウトラインがそれぞれほぼ円形になるまで平滑化を行えば、自己交錯を起こすことなくモルフィングを行うことができる。しかし、対象図形の形状に関わらずモルフィングの際にアウトラインが円形に近づいてしまうのは、視覚的に好ましくない。そこで、実際には、変形前後の二つの対象図形におけるアウトラインに対してそれぞれ数回ラプラシアン・スムージングを適用し、円形に近づく前の途中の段階で平滑化処理をうち切り、適度に平滑化された二つのアウトラインに対してモルフィングを行う。なお、どの段階で平滑化をうち切るかを定めるには、例えば、試行的にモルフィングを行った結果をフィードバックして、自己交錯が起きなくなるまでラプラシアン・スムージングを繰り返す方法を取ることができる。すなわち、モルフィングを試行して自己交錯が起きないかどうかを確認し、自己交錯が起きたならばラプラシアン・スムージングを行って再びモルフィングを試行する。そして、自己交錯が起きなければ、その時点で平滑化処理を止める。以上のような処理を行った後、アウトライン編集部50は、二つの対象図形にそれぞれ対応する二つのアウトラインの集合を出力する。当該アウトラインの集合には、アウトライン編集部50に入力された初期段階のアウトラインと、ラプラシアン・スムージングを施すことにより得られたアウトラインとが含まれる。ただし、ラプラシアン・スムージングにより得られたアウトラインを全て出力する必要はなく、得られたアウトラインの数に応じて、適当な段階の平滑化されたアウトラインを数個出力すれば良い。また、最初にラプラシアン・スムージングを行わずにモルフィングを試行した時点で自己交錯が起きなかった場合は、平滑化処理を行う必要はない。

い。この場合は、アウトライン編集部50に入力された初期段階の二つのアウトラインが出力されることとなる。

【0040】シーケンス生成部60は、アウトライン木生成部20から出力されたアウトライン木とアウトライン編集部50から出力されたアウトラインの集合とを用いて、モルフィング処理全体のタイムパラメータに沿った当該変形処理の経過におけるアウトラインのシーケンスファイルを生成する。すなわち、当該シーケンスファイルには、当該変形処理の各段階におけるアウトラインがタイムパラメータの順に並べられている。シーケンス生成部60により生成されたアウトラインのシーケンスファイルは、シーケンスファイル格納部70に格納される。

【0041】モルフィング・ビューア80は、シーケンスファイル格納部70からシーケンスファイルを読み出して、当該シーケンスファイルのアウトラインを当該タイムパラメータに沿って再生しながら、各アウトラインの間をモルフィングにより滑らかに変形させる動画をディスプレイ装置に表示する。これにより、ディスプレイ装置には、一方の対象図形から他方対象図形へ、自己交錯を起こすことなく滑らかに輪郭が変形する様子が表示されることとなる。なお、ここではアウトラインのシーケンスファイルをシーケンスファイル格納部70から読み出すこととしたが、シーケンス生成部60から出力されたシーケンスファイルを直接入力するようにしても良い。

【0042】次に、本実施の形態の動作について説明する。上述したように、本実施の形態は、モルフィングによる変形処理の対象図形として任意の形状の図形に対して柔軟に対応することができる。ここでは、文字を対象図形とした場合を例として説明する。また、対象図形である文字は、コンピュータでディスプレイ表示やプリントアウトに一般的に用いられるアウトライン・フォントであるものとする。なお、文字を対象図形とする場合、対象図形である文字がドット・フォントであっても、前処理として当該文字の輪郭線を抽出する処理が実現できれば本実施の形態による変形処理を行うことが可能である。また、本動作例では、二つの文字を1対1で対応させて一方から他方へ変形させる場合について説明するが、 $m$ 個の文字列と $n$ 個の文字列を $m$ 対 $n$ で対応させて変形させることも可能である。このような多対多で変形処理を行う場合に必要となる特殊な処理については後述する。

【0043】図2は、本実施の形態のモルフィング処理装置による全体的な処理の流れを説明するフローチャートである。図2を参照すると、まず、アウトライン抽出部10により、フォントファイルから対象図形として選択された二つの文字のフォントデータが読み出される。そして、当該二つの文字のアウトラインが抽出される。

(ステップ201)。ここで、当該文字のアウトライン・フォントが、パーソナル・コンピュータでディスプレイ表示や印刷に広く用いられているTrue Type フォントである場合、アウトラインの抽出は、例えばOpenGLにおける`glUseFontOutline`関数やWindows (マイクロソフト社の商標) オペレーティングシステム に用意されている`GetGlyphOutline`関数等の既存のAPI (Application Program Interface) を用いて行うことができる。この種の関数を用いれば、所定のインデックスを与えると、対応する文字のアウトライン構造として、アウトラインの数、当該文字に含まれる点 (アウトラインを構成する点)、当該点の座標列 (ベクトル列)、当該点に付随するフラグ列、アウトラインの終端点列等を得ることができる。点に付随するフラグ列には、当該点が線分の端点であるか2次元のベジェ曲線の制御点であるかを識別する情報が含まれる。True Type フォントでは、線分の真っ直ぐな部分は線分で、滑らかな曲線部分は2次元のベースワイス・ベジェ曲線 (piecewise Bezier curves) で表現されている。アウトライン抽出部10は、このようにして文字のアウトラインを取得し、かつベジェ曲線で表現された部分を線分近似して出力する。なお、文字の輪郭は複数のアウトラインで形成される場合が多いので、アウトライン抽出部10の出力はアウトラインの集合となる (図1の{F×1}、{F×2}参照)。なお、ここでは対象図形がTrue Type フォントである場合について説明したが、Adobe社のType 1 フォントなどの他のアウトライン・フォントに関しても固有のAPIを用いればアウトラインの抽出が可能である。

【0044】次に、アウトライン木生成部20により、対象図形である文字のアウトライン木が生成される (図2、ステップ202)。図3にアウトライン木生成部20による処理の詳細なアルゴリズムを示す。図3を参照すると、まず、アウトライン抽出部10により抽出されたアウトラインの集合{F×1}、{F×2}を入力して (ステップ301)、各文字の輪郭を形成する各アウトラインに対してバウンディング・ボックスを設定し、その面積を計算する (ステップ302)。次に、アウトライン全体のバウンディング・ボックスを生成し、当該バウンディング・ボックスに対応させてアウトライン木のルート・ノードを生成する (ステップ303)。ここで、アウトライン全体のバウンディング・ボックスとは、例えば漢字「杉」のように複数のバウンディング・ボックスが並べて配置される関係にある場合に、それらを全て包含するバウンディング・ボックスという意味である。したがって、文字「B」のように最外郭のアウトラインに対応するバウンディング・ボックスが当該文字のアウトラインの全部を包含する場合は、当該バウンディング・ボックスとアウトライン全体のバウンディング・ボックスとは同一となる。また、ルート・ノードには、バウンディング・ボックスに関する情報 (例えば、

図8に示す情報のうち、 $x_{min}$ ,  $x_{max}$ ,  $y_{min}$ ,  $y_{max}$ と、 $ex$ ,  $ey$ ) のみが格納される。

【0045】次に、設定されたバウンディング・ボックスを、面積の大きいものから順にソートしてリストを作成する (ステップ304)。次に、当該リストの先頭のバウンディング・ボックスのアウトラインに着目して、当該アウトラインに対応する子ノードを生成する (ステップ305)。そして、当該子ノードをアウトライン木に挿入する (ステップ306)。ここで、当該子ノードにおけるアウトラインのバウンディング・ボックスが、既にアウトライン木に挿入されている他の子ノードにおけるアウトラインのバウンディング・ボックスに完全に含まれる場合、当該子ノードを当該他の子ノードの下に再帰的に挿入する。次に、当該リストの要素が一つ (最後にアウトライン木に挿入された子ノードに対応するバウンディング・ボックス) だけかどうかを確認し、要素が一つだけであれば処理を終了してアウトライン木を出力する (ステップ307、309)。一方、当該リストの要素が複数ある場合は、先頭の要素 (最後にアウトライン木に挿入された子ノードに対応するバウンディング・ボックス) を削除し、新しい先頭のバウンディング・ボックスのアウトラインに着目して、同様の処理を行う (ステップ307、308)。以上のようにして、アウトライン木生成部20から対象図形である二つの文字のアウトライン木 $M \times 1$ ,  $M \times 2$ が出力される。

【0046】次に、アウトライン木整理部30により、アウトライン木の整理が行われる (図2、ステップ203)。図4にアウトライン木整理部30による処理の詳細なアルゴリズムを示す。なお、ここでは、アウトライン木の整理として、面積が最大のバウンディング・ボックスのみを残すこととする。図4を参照すると、まず、アウトライン木生成部20により生成されたアウトライン木 $M \times 1$ ,  $M \times 2$ を入力する (ステップ401)。そして、ルート・ノードと、ルート・ノードの直下にあるノードのうちで最大の面積の子ノードにおける`delete flag`をオフとし、他の子ノードにおける`delete flag`をオンとする (ステップ402)。次に、`delete flag`がオンである子ノードをバウンディング・ボックスの中心に向けて移動させ、最終的にディスプレイ装置に表示しない点とすることにより行う。次に、以上のようにして残された単一の子ノードのアウトラインを出力する (ステップ403)。以上のようにして、アウトライン木整理部30から、対象図形である二つの文字に関して一つずつ、面積が最大のバウンディング・ボックスのアウトライン $OL1$ ,  $OL2$ が出力される。

【0047】次に、頂点調整部40により、アウトライン $OL1$ ,  $OL2$ の頂点の数を一致させる処理が行われる (図2、ステップ204)。図5に、頂点調整部40による処理の詳細なアルゴリズムを示す。図5を参照すると、まず、アウトライン $OL1$ ,  $OL2$ を入力し (ス

ステップ501)、それぞれの頂点列を、Y座標の値が最大の頂点(複数ある場合はその中でX座標の値が最大の頂点)を先頭にして並べ替える(ステップ502)。ここでは、二つのアウトラインOL1、OL2の頂点列の先頭がバウンディング・ボックスのほぼ右上に揃うように並べ替えを行ったが、上述したように、先頭の位置をバウンディング・ボックスのどこに設定しても良い。次に、アウトラインOL1、OL2のそれぞれの全周の長さを求め、各頂点の座標値を $[0, 0 - 1, 0]$ のパラメータで表現する(ステップ503)。次に、一方のアウトラインにおけるパラメータ化された頂点を用いて、他方のアウトラインに頂点を線形補間する操作を、相互に行う(ステップ504)。これにより、二つのアウトラインにおける頂点の数が一致する。次に、頂点の数が同じになった二つのアウトラインOL1'、OL2'を出力する(ステップ505)。

【0048】次に、アウトライン編集部50により、アウトラインの編集が行われる(図2、ステップ205)。この時、必要に応じてアウトラインへの平滑化処理も行われる。図6にアウトライン編集部50による処理の詳細なアルゴリズムを示す。なお、ここでは平滑化処理としてラブラシアン・スムージング処理を行うこととする。図6を参照すると、まず、アウトラインOL1'、OL2'を入力し(ステップ601)、二つのアウトラインに対してモルフィングを行う(ステップ602、603)、自己交錯が発生しないかどうかを調べる(ステップ604)。そして、自己交錯が発生した場合は、各アウトラインにラブラシアン・スムージングを行う(ステップ605)。次に、ラブラシアン・スムージングを施されたアウトラインOL1(i)'、OL2(j)'に対して、再びモルフィングを行う(ステップ606、603)。以下、自己交錯が発生しなくなるまで同様の処理を繰り返す。ここで、OL1(i)'は、i回のラブラシアン・スムージングを施されたアウトラインOL1'であることを示す。同様に、OL2(j)'は、j回のラブラシアン・スムージングを施されたアウトラインOL2'であることを示す。また、パラメータmは、ステップ605において、アウトラインOL1(i)'に対してm回のラブラシアン・スムージングを施すことを示し、パラメータnは、アウトラインOL2(j)'に対してn回のラブラシアン・スムージングを施すことを示す。また、ステップ606のパラメータm、nを更新する式、 $m = m + p$  ( $p \geq 1$ )、 $n = n + q$  ( $q \geq 1$ )において、パラメータp及びパラメータqを任意に設定することができる。これにより、モルフィングにおいて自己交錯が発生した場合に、各アウトラインに対して何回ずつラブラシアン・スムージングを施すかを設定することができる。

【0049】以上のようにして、モルフィングを行った際に自己交錯が発生しなくなったならば、アウトライン

OL1'、OL2'のそれぞれに関して、ラブラシアン・スムージングを行ったことにより得られたいくつかのアウトラインの集合{OL1'}、{OL2'}を出力する(ステップ604、607)。なお、アウトラインOL1'、OL2'に対して複数回のラブラシアン・スムージングを行った場合、必ずしもラブラシアン・スムージングを行って得られたアウトラインの全てを出力する必要はなく、得られたアウトラインを数個おきに取り出して出力するようにしても良い。また、入力したアウトラインOL1'、OL2'が、ラブラシアン・スムージングを施すことなく、そのまま自己交錯を起こさずにモルフィングできた場合(1回目のステップ604の判断で自己交錯が発生しなかった場合)、出力されるアウトラインの集合の要素は、{OL1'}の要素がアウトラインOL1(0)' (= OL1') だけであり、{OL2'}の要素がアウトラインOL2(0)' (= OL2') だけである。

【0050】次に、シーケンス生成部60により、変形処理全体におけるアウトラインのシーケンスファイルが生成される(図2、ステップ206)。図7にシーケンス生成部60による処理の詳細なアルゴリズムを示す。図7を参照すると、まず、アウトライン生成部20から出力されたアウトライン木X1、X2と、アウトライン編集部50から出力されたアウトラインの集合{OL1'}、{OL2'}とを入力する(ステップ701)。ここでは、アウトラインの集合{OL1'}にはr個のアウトラインが含まれ、アウトラインの集合{OL2'}にはs個のアウトラインが含まれるものとする。次に、アウトライン木X1が2個以上の子ノードを持つかどうかを調べる(ステップ702)。アウトライン木X1に2個以上の子ノードがある場合は、時刻t1を決めて、 $[t0, t1]$ の間に図8に示したdelete flagがオンとなっている子ノードを消滅させる処理を行うこととし、当該アウトラインのシーケンスを作成する(ステップ703)。これにより、アウトライン木X1に対応する対象図形に関して、最初に対象図形から抽出された状態のアウトライン(t0に対応)と、アウトライン木X1を整理して残された子ノードに対応するアウトラインだけで構成されるアウトライン(t1に対応)の二つのアウトラインについてのシーケンスが作成されることとなる。

【0051】次に、アウトライン木X1の子ノードが1個である場合(ステップ702でYesの場合)、またはステップ703でアウトライン木X1の子ノードを1個に整理した後(ステップ702でNoの場合)、アウトラインの集合{OL1'}の要素の数に合わせて、r個のタイムパラメータ $t_i, t_{i+1}, \dots, t_{i+r-1}$ におけるアウトラインのシーケンスを作成する(ステップ704)。ここで、各タイムパラメータの関係は、 $t_i \leq t_{i+1} \leq \dots \leq t_{i+r-1}$ である。また、アウトライン木X1の

子ノードが元々1個であった場合(ステップ701でYes)は、 $i=1$ である。そして、アウトライン木 $\times 1$ の子ノードを1個に整理した場合(ステップ701でNo)は、 $i=2$ である。次に、アウトラインの集合 $\{OL2'\}$ の要素の数に合わせて、 $s$ 個のタイムパラメータ $t_{itr}, t_{itr+1}, \dots, t_{itr+s-1}$ におけるアウトラインのシーケンスを作成する(ステップ705)。ここで、各タイムパラメータの関係は、 $t_{itr} \leq t_{itr+1} \leq \dots \leq t_{itr+s-1}$ である。次に、アウトライン木 $\times 2$ が2個以上の子ノードを持つかどうかを調べる(ステップ706)。アウトライン木 $\times 2$ に2個以上の子ノードがある場合は、時刻 $t_{itr+q}$ を決めて、 $[t_{itr+q}, t_n]$ の間に図8に示したdelete flagがオンとなっている子ノードを消滅させる処理の逆プロセスを行うこととし、当該アウトラインのシーケンスを作成する(ステップ707)。

ここで、 $t_n=1, 0$ である。これにより、アウトライン木 $\times 2$ に対応する対象図形に関して、アウトライン木 $\times 2$ を整理して残された子ノードに対応するアウトラインだけで構成されるアウトライン( $t_{itr+q}$ に対応)と、最初に対象図形から抽出された状態のアウトライン( $t_n$ に対応)の二つのアウトラインについてのシーケンスが作成されることとなる。以上の処理により、時刻 $t_0, t_1, \dots, t_n (t_0 \leq t_1 \leq \dots \leq t_n)$ におけるアウトラインのシーケンスファイルが作成され、出力される(ステップ708)。出力されたシーケンスファイルは、シーケンスファイル格納部70に格納される。

【0052】次に、モルフィング・ビューア80により、シーケンスファイルに基づいてモルフィングが実行される(図2、ステップ207)。すなわち、シーケンスファイル格納部70からシーケンスファイルを読み出して、タイムパラメータ $t_0, t_1, \dots, t_n$ に沿って再生しながら、各アウトラインの間をモルフィングにより滑らかに変形させる動画をディスプレイ装置に表示する。これにより、アウトライン木 $\times 1$ に対応する文字

(対象図形)からアウトライン木 $\times 2$ に対応する文字(対象図形)へ、自己交錯を起こすことなく滑らかに輪郭が変形する様子が表示される。反対に、アウトライン木 $\times 2$ に対応する文字からアウトライン木 $\times 1$ に対応する文字へ変形させる場合は、タイムパラメータを $t_0, t_1, \dots, t_n$ 逆に辿って再生し、各アウトラインの間をモルフィングにより滑らかに変形させる動画をディスプレイ装置に表示する。なお、モルフィング・ビューア80が、シーケンスファイルにおける各アウトラインの間を滑らかに変形させる動画を作成してディスプレイ装置に表示する処理については、従来の一般的なモルフィングによる画像処理と同様であるため、説明を省略する。

【0053】以上のように、本実施の形態によれば、対象図形のアウトラインを線分列として取り出し、適宜、平滑化処理を行ってからモルフィングを行うため、処理の対象となる図形の形状に制限がない。更に、対話的に

対象図形のアウトラインにおけるサンプル点を対応付けさせる必要もない。このため、汎用的に利用することができるものである。特に、文字のような複雑な形状の図形に対しても自己交錯の起こらないモルフィングを実現でき、アウトライン・フロント等のように文字のアウトラインを容易に抽出できるのであれば、ディスプレイ装置に表示されている文字をモルフィングにより変形させることができ、きわめて視覚的効果の高い文書を作成することができる。

【0054】次に、本実施の形態による具体的な実施例を説明する。図12は本実施の形態により文字「S」を文字「C」に変形した際の経過を示す図である。この例では、文字「S」、文字「C」のいずれも単一のアウトラインで形成されているため、アウトライン木整理部30による処理は行われない。図中、12-1から12-4までの間に、文字「S」に対して数回のラブラシアン・スムージングが施されている。また、12-8から12-6までの間に、文字「C」に対して数回のラブラシアン・スムージングが施されている。実際のディスプレイ表示では、12-1から12-8までの変形、及びその逆プロセスである12-8から12-1までの変形が、各状態の間を滑らかな動作で移行する動画として表示される。

【0055】図13は本実施の形態により文字「B」を文字「A」に変形した際の経過を示す図である。図中、13-1、13-2は、文字「B」のうち、面積最大のバウンディング・ボックスに対応するアウトライン、すなわち最外郭のアウトラインだけを残して、他のアウトラインを消滅させる様子を示す。また、13-3、13-4は、残された最外郭のアウトラインに対してラブラシアン・スムージングを施したことによる変形の様子を示す。同様に、13-7、13-8は、文字「A」のうち、最外郭のアウトラインだけを残して他のアウトラインを消滅させる処理の逆プロセスを示す。また、13-5、13-6は、残された最外郭のアウトライン(13-7の状態)に対してラブラシアン・スムージングを施したことによる変形の逆プロセスを示す。実際のディスプレイ表示では、13-1から13-8までの変形、及びその逆プロセスである13-8から13-1までの変形が、各状態の間を滑らかな動作で移行する動画として表示される。

【0056】図14は本実施の形態により文字「人」を文字「山」に変形した際の経過を示す図である。この例では、文字「人」、文字「山」のいずれも単一のアウトラインで形成されているため、アウトライン木整理部30による処理は行われない。図中、14-1から14-3までの間に、文字「人」に対して数回のラブラシアン・スムージングが施されている。また、14-8から14-4までの間に、文字「山」に対して数回のラブラシアン・スムージングが施されている。実際のディスプレ



イ表示では、14-1から14-8までの変形、及びその逆プロセスである14-8から14-1までの変形が、各状態の間を滑らかな動作で移行する動画として表示される。

【0057】図15は本実施の形態により文字「青」を文字「杉」に変形した際の経過を示す図である。図中、15-1から15-4までは、文字「青」のうち、面積最大のバウンディング・ボックスに対応するアウトラインだけを残して、他のアウトラインを消滅させる様子を示す。ここでは、文字「青」の上半分が残り、下の月の部分が消滅される。また、15-5、15-6は、残された最外部のアウトラインに対してラブラシアン・スムージングを施したことによる変形の様子を示す。同様に、15-10から15-12までは、文字「杉」のうち、面積最大のバウンディング・ボックスに対応する木偏の部分のアウトラインだけを残して他のアウトラインを消滅させる処理の逆プロセスを示す。また、15-7から15-9までは、残された木偏部分のアウトライン（15-10の状態）に対してラブラシアン・スムージングを施したことによる変形の逆プロセスを示す。実際のディスプレイ表示では、15-1から15-12までの変形、及びその逆プロセスである15-12から15-1までの変形が、各状態の間を滑らかな動作で移行する動画として表示される。

【0058】図16は本実施の形態をフォントではなく一般的な閉曲線に適用したものである。図示の例では、犬の輪郭形状を模した図形（アウトライン）から鳥の輪郭形状を模した図形（アウトライン）に変形した際の経過を示す図である。この例では、いずれの図形も単一のアウトラインで形成されているため、アウトライン木整理部30による処理は行われない。図中、16-1から16-4までの間に、犬の図形のアウトラインに対して数回のラブラシアン・スムージングが施されている。また、16-8から16-5までの間に、鳥の図形のアウトラインに対して数回のラブラシアン・スムージングが施されている。実際のディスプレイ表示では、16-1から16-8までの変形、及びその逆プロセスである16-8から16-1までの変形が、各状態の間を滑らかな動作で移行する動画として表示される。

【0059】上記の動作例及び実施例では、二つの対象図形を1対1で対応させて一方から他方へ変形させる場合について説明したが、本実施の形態を用いて、 $m$ 個の文字列と $n$ 個の文字列を $m$ 対 $n$ で対応させて変形させることも可能である。この場合、任意の $m$ 個の文字を要素とする文字列と任意の $n$ 個の文字を要素とする文字列とを処理対象として入力し、当該二つの文字列の要素である各文字を1対1で対応させる。任意の $m$ 個の文字列と任意の $n$ 個の文字列であるから、二つの文字列において文字の数が異なる場合がある。この場合は、文字が不足している方の文字列に、不足している文字の数だけ、デ

イスプレイ装置に表示されない円を設けて、他方の文字列における余っている文字に対応させる。そして、対応付けられた要素ごとに、上述した1対1の場合と同様にモルフィングを行う。このようにすれば、一方の文字列から他方の文字列へ滑らかに変形させることができる。

【0060】文字とディスプレイ装置に表示されない円との間の変形については、全般的な処理は文字どうしの変形の場合と同様である。ただし、円に対してはラブラシアン・スムージングを施す必要はない。また、頂点調整部40による処理において、文字（対象図形）のアウトラインにおける頂点に対応する点を当該円に追加する場合、次のようにして点の座標を求める。すなわち、文字のアウトラインの頂点を特定するパラメータに対応するパラメータとして、下記の数4に示すパラメータ $\theta$ を設定する。

【数4】

$$\theta = p \times 2 \times \pi$$

そして、当該パラメータ $p$ で特定された頂点に対応する当該円における点の座標（ $x_p$ ,  $y_p$ ）を、当該円の半径を $r$ 、中心座標を（ $x_0$ ,  $y_0$ ）として、下記の数5で求める。

【数5】

$$x_p = x_0 + r \times \sin \theta$$

$$y_p = y_0 + r \times \cos \theta$$

以上のようにして、円の上に設けられた点と文字のアウトラインの頂点とが対応付けられるので、モルフィングを行うことが可能となる。

【0061】実際のディスプレイ表示では、文字どうしが対応付けられている箇所では、一方の文字から他方の文字へ変形する。また、文字と上述したディスプレイ装置に表示されない円とが対応付けられている箇所では、文字が次第に小さくなって消えたり、何もないスペースから文字が現れるように表示されることとなる。これは、視覚的には、上述した複数のアウトラインで構成される文字において、一つのアウトラインを残して他のアウトラインを消滅させる場合（図13及び図15参照）と同様である。なお、ここでは複数の文字を含む文字列どうしをモルフィングにより変形させる場合について説明したが、文字以外の任意の図形を要素として含む図形列を処理対象とすることもできる。

【0062】以上の動作例及び実施例では、二次元の図形を対象としてモルフィングを行う方法について説明したが、本実施の形態は、3次元画像に対しても応用することができる。すなわち、3次元画像を、ポリゴン等を用いて近似し、表面を線分列で表現すれば、本実施の形態による平滑化処理を行って、自己交錯のないモルフィング処理を行うことが可能となる。

# 【0063】

【発明の効果】以上説明したように、本発明によれば、多種多様な図形の輪郭線に対してベクトル空間でモルフィングによる画像処理ができるようにすることができる。具体的には、複数の輪郭線で形成された図形に対してもモルフィングを行うことができる。また、凹凸の多い図形に対しても自己交錯を起こさずモルフィングを行うことができる。

【0064】また、コンピュータのディスプレイ装置上に表示された文字や文字列に対してもモルフィングによる画像処理を行うことができる。

## 【図面の簡単な説明】

【図1】 本実施の形態におけるモルフィング処理装置の全体構成を説明するための図である。

【図2】 本実施の形態のモルフィング処理装置による全体的な処理の流れを説明するフローチャートである。

【図3】 アウトライン木生成部20による処理の詳細なアルゴリズムを説明するフローチャートである。

【図4】 アウトライン木整理部30による処理の詳細なアルゴリズムを説明するフローチャートである。

【図5】 頂点調整部40による処理の詳細なアルゴリズムを説明するフローチャートである。

【図6】 アウトライン編集部50による処理の詳細なアルゴリズムを説明するフローチャートである。

【図7】 シーケンス生成部60による処理の詳細なアルゴリズムを説明するフローチャートである。

【図8】 アウトライン木のノードに格納されるデータを例示して説明する図である。

【図9】 文字「B」を対象図形とした場合のアウトライン木を説明する図である。

【図10】 文字「杉」を対象図形とした場合のアウトライン木を説明する図である。

【図11】 アウトラインを消滅させる方法をC言語風に記述したものである。

【図12】 文字「S」を文字「C」に変形した際の経過を説明する図である。

【図13】 文字「B」を文字「A」に変形した際の経過を説明する図である。

【図14】 文字「人」を文字「山」に変形した際の経過を説明する図である。

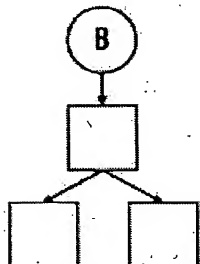
【図15】 文字「青」を文字「杉」に変形した際の経過を説明する図である。

【図16】 犬の輪郭形状を模したアウトラインを鳥の輪郭形状を模したアウトラインに変形した際の経過を説明する図である。

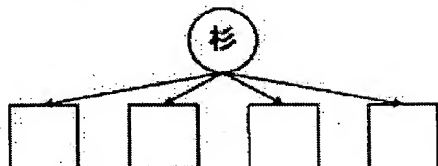
## 【符号の説明】

10…アウトライン抽出部、20…アウトライン木生成部、30…アウトライン木整理部、40…頂点調整部、50…アウトライン編集部、60…シーケンス生成部、70…シーケンスファイル格納部、80…モルフィング・ビューア

【図9】



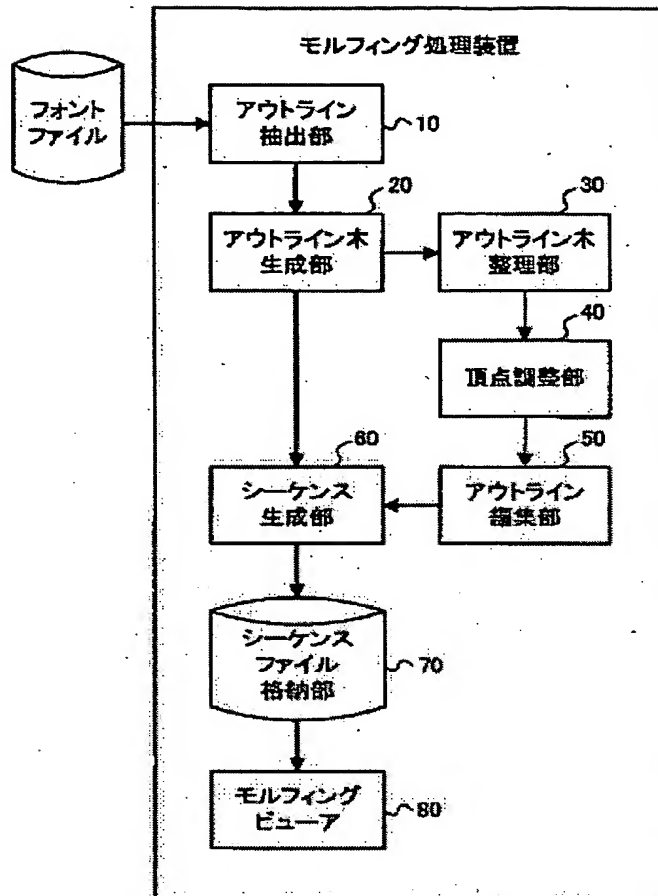
【図10】



【図11】

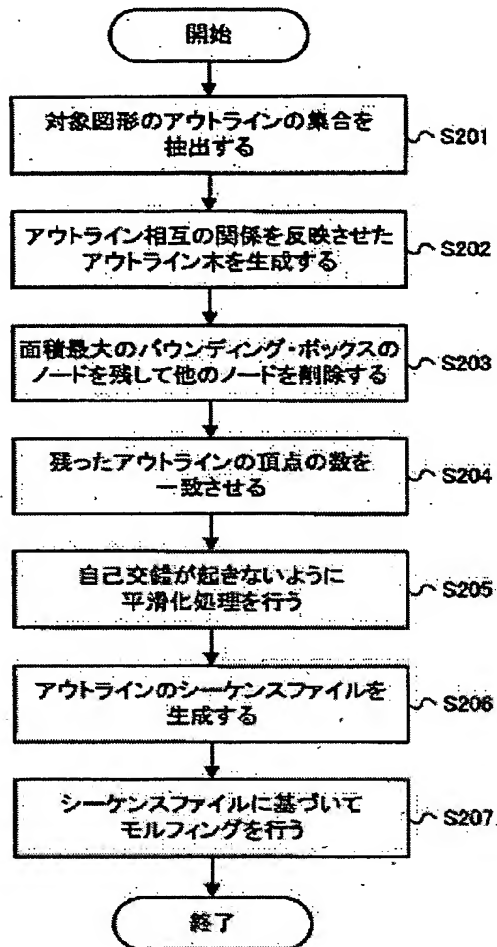
```
u=(t-T1)/(T2-T1);
v=(T2-t)/(T2-T1);
for(i=0; i<node->child[k]->numPoint; i++){
    x=v*node->child[k]->outline[i][0]+u*node->child[k]->cx;
    y=v*node->child[k]->outline[i][1]+u*node->child[k]->cy;
}
```

【図1】



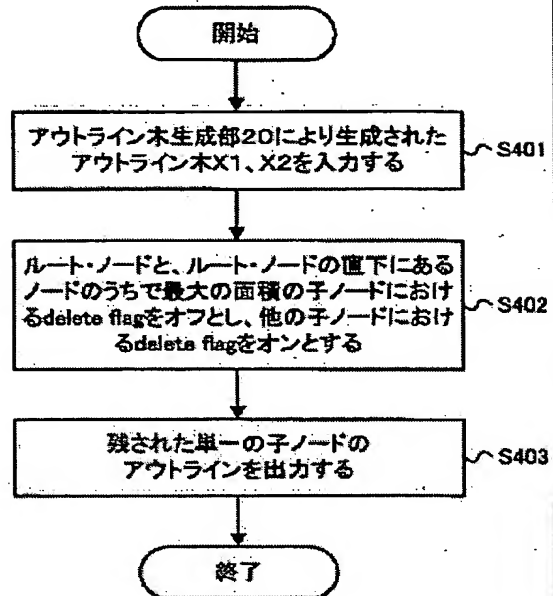
【図 2】

全体的な処理の流れ

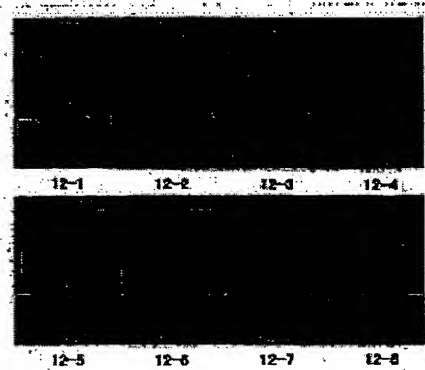


【図 4】

アウトライン木整理部30による処理

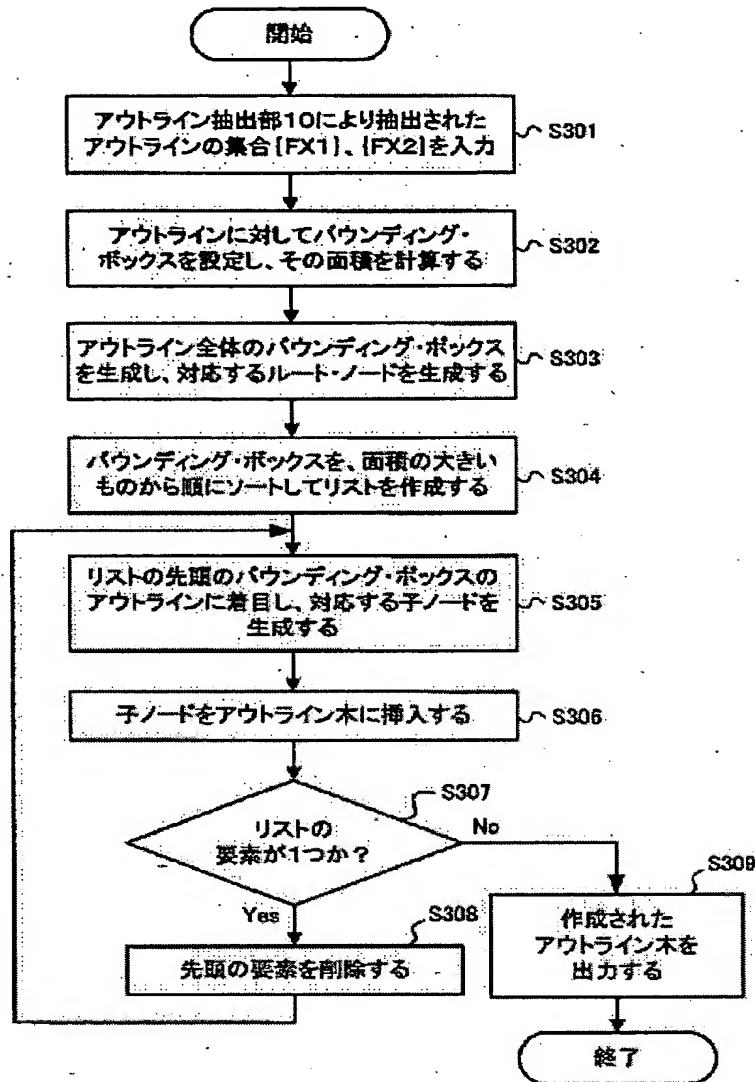


【図 12】



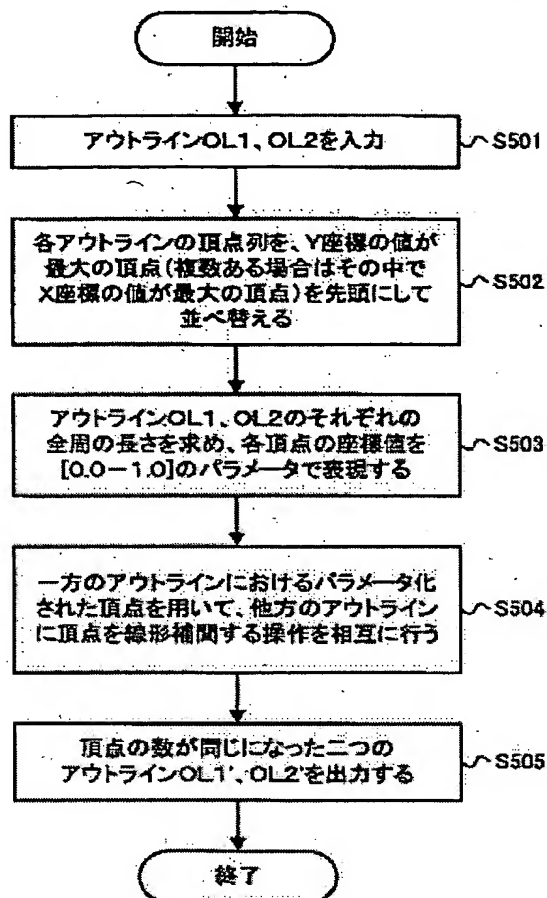
【図 3】

アウトライン木生成部20による処理

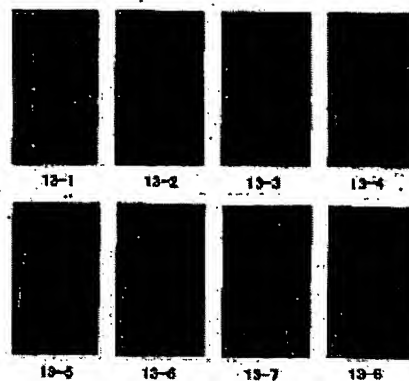


【図 5】

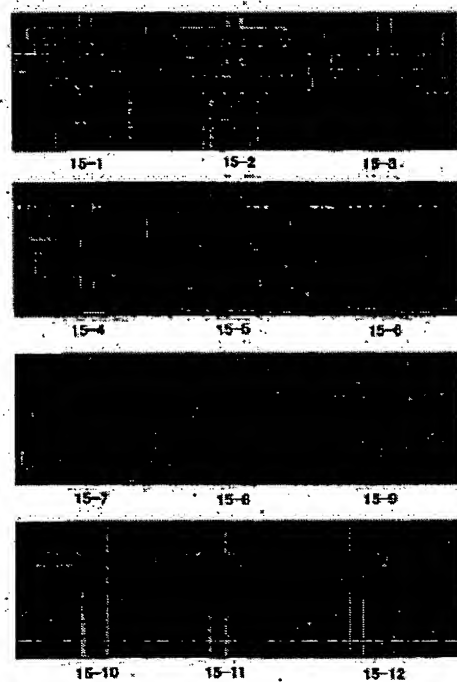
頂点調整部40による処理



【図 13】

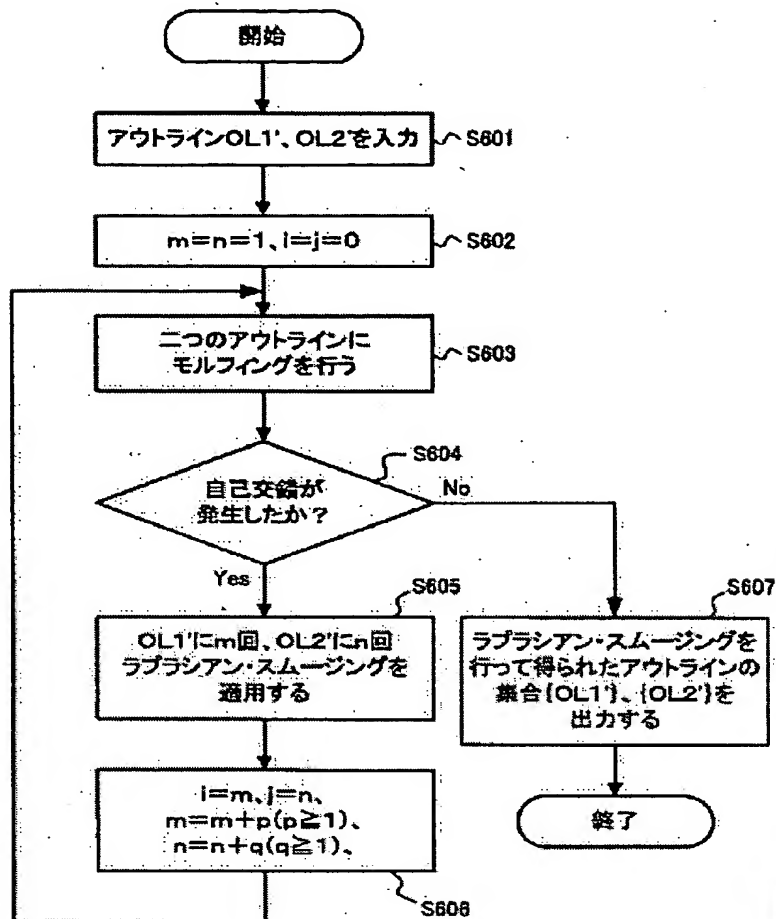


【図 15】



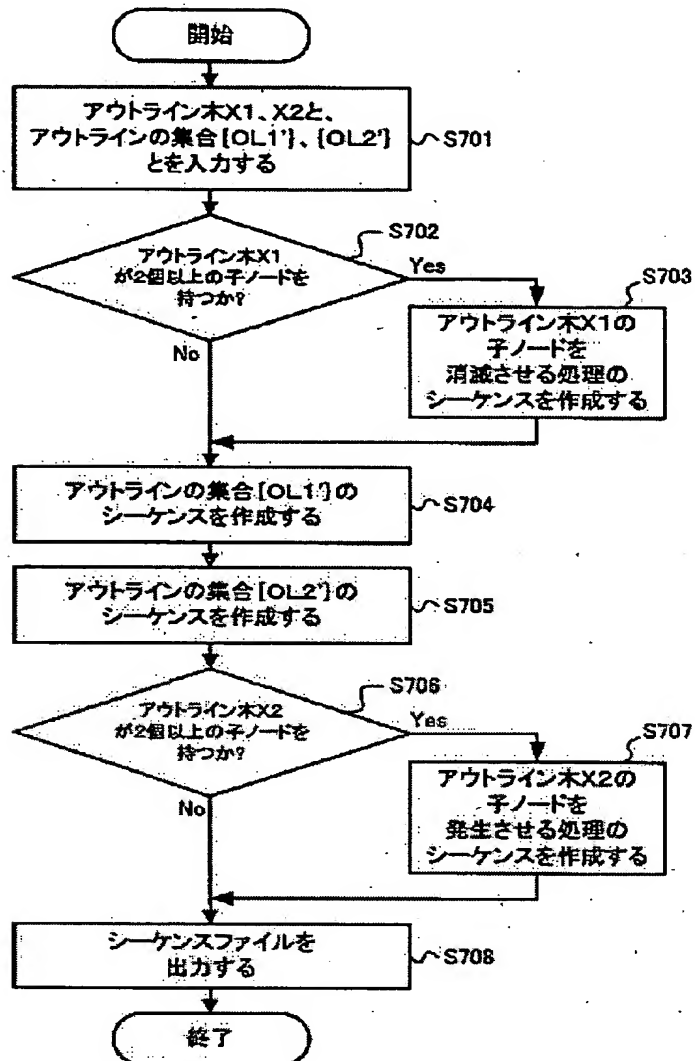
【図6】

アウトライン編集部50による処理



【図7】

シーケンス生成部60による処理



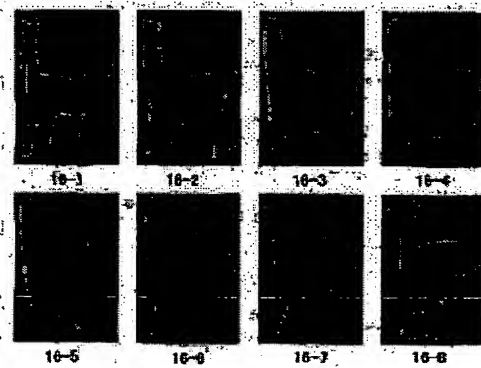


【図 8】

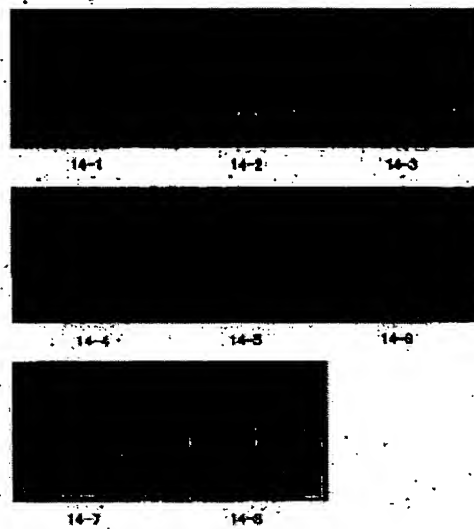
アウトライン木のノードの中身のデータ

delete flag(整数)	アウトライン木の整理の際に削除 するかどうかを設定するフラグ
num Point(整数)	アウトラインの頂点の数
num Child(整数)	自ノードのしたに位置する 子ノードの数
outline(実数配列)	アウトラインの各頂点における 初期の座標値
deform outline(実数配列)	アウトラインの各頂点における モルフィング時に使用される座標値
xmin, xmax, ymin, ymax(実数)	バウンディング・ボックスの座標
child(ノードへのポインタ配列)	自ノードの子ノードへのポインタ
cx, cy(実数)	バウンディング・ボックスの 中心の座標値

【図 16】



【図14】



フロントページの続き

(72)発明者 青野 雅樹  
神奈川県大和市下鶴間1623番地14 日本ア  
イ・ビー・エム 株式会社 東京基礎研究所  
内

Fターム(参考) 5B050 BA08 EA06 EA13 EA15 EA24

**This Page is Inserted by IFW Indexing and Scanning  
Operations and is not part of the Official Record**

**BEST AVAILABLE IMAGES**

Defective images within this document are accurate representations of the original documents submitted by the applicant.

Defects in the images include but are not limited to the items checked:

- ☐ **BLACK BORDERS**
- ☐ **IMAGE CUT OFF AT TOP, BOTTOM OR SIDES**
- ☐ **FADED TEXT OR DRAWING**
- ☐ **BLURRED OR ILLEGIBLE TEXT OR DRAWING**
- ☐ **SKEWED/SLANTED IMAGES**
- ☒ **COLOR OR BLACK AND WHITE PHOTOGRAPHS**
- ☐ **GRAY SCALE DOCUMENTS**
- ☐ **LINES OR MARKS ON ORIGINAL DOCUMENT**
- ☐ **REFERENCE(S) OR EXHIBIT(S) SUBMITTED ARE POOR QUALITY**
- ☐ **OTHER:** \_\_\_\_\_

**IMAGES ARE BEST AVAILABLE COPY.**

**As rescanning these documents will not correct the image problems checked, please do not report these problems to the IFW Image Problem Mailbox.**